**International Journal of Information and Computer Science**
IJICS Volume 1, Issue 2, May 2012 PP. 16-20
ISSN(online) 2161-5381
ISSN(print) 2161-6450
www.iji-cs.org

# Improved Genetic Algorithm for Dynamic Path Planning

Xuan Zou[1], Bin Ge[1], Peng Sun[1]
[1]College of information & Engineering, Dalian University, Dalian, China
Email: ichzx@126.com, gebin58@163.com,pengsun007@sina.com

**(Abstract)** Navigation technology is key part in robot technology and the robotics, while path planning is one of the most important branches in robot navigation. Secure obstacle avoidance based on reasonable algorithms can be achieved by using a variety of sensors to collect environmental information. A new method integrating various of path planning approaches has been proposed in this paper to optimized robots' moving path. Then it can improve their adaptive capacity. On this basis, mathematical optimization selection of fitness function in robot path planning with the help of improved genetic algorithm can raise the efficiency of arithmetic operations.

**Keywords:** Improved Genetic Algorithms; Path Planning; Grid Method; Fitness Function

## 1. INTRODUCTION

Mobile robot path planning is a combined research of artificial intelligence and robotics. Scholars from all over the world have carried out a lot of studies on robot path planning by using the artificial potential field method, neural network method, ant colony algorithm, genetic algorithm and so on. Artificial potential field method is suitable for underlying real-time control with the lack of global information. But it has the problem of local optimum[1]. Neural network method has good ability to learn. The network structure becomes large when considering lots of obstacles or dynamic environment, and the neuronal threshold constantly changes as time goes by[2]. Ant colony algorithm can achieve global optimization while its speed of convergence is slow, and it is more sensitive to initial parameter setting[3]. Genetic algorithm is a computational model based on the genetic selection of Darwin and biological evolution process of natural selection. It was first proposed by J.Holland Professor, the University of Michigan in USA in 1975. It is a kind of search algorithm for optimization of complex systems. And it is robustness. There are features of multi-point search and probability of search. Genetic algorithm needs only to direct the search direction of the objective function and the corresponding fitness function as the search. It does not require derivative and other auxiliary knowledge. Therefore, the genetic algorithm provides a general framework for solving complex system problems. It does not depend on the specific areas of the problem. It is also robustness types of problems[4].

Path planning optimization goal generally include the shortest path length, minimum energy consumption and the shortest time. Therefore, the mobile robot path planning essentially can be viewed as a conditional constrained optimization problem. Environmental model is a direct result of the algorithm for computing difficult throughout the planning process. The more environmental of the detailed division, the more initial information it can get. The analysis is more accurate than before. But the operation will be difficult. However, path planning will be precise. We should be based on real-time adjustment to the environment in order to make robot adaptive capacity. The schematic diagram of robot path planning is shown in **Figure 1**.

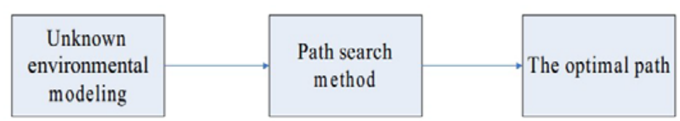


**Figure 1.** Robot path planning schematic

Robot path planning proposed a new algorithm based on genetic algorithms in this paper. It divided into two main areas which are optimization of environmental modeling and mathematical optimization of the fitness function processing. These practices are a combination of genetic algorithms and robot path object's characteristics. It guarantees that the algorithm can fast convergence. Finally the planning process becomes more simple and effective.

## 2. OPTIMIZATION OF ENVIRONMENTAL MODELING

Robot can sense the environment and their own state by all kinds of sensors. It can be achieved a voluntary movement of the goal-oriented environment with obstacles. Path planning method can get satisfactory results in a completely known environment from the present study. Unknown environment, in particular, may lose their effectiveness in obstacle unfixed position. In this paper, the genetic algorithm can simulate the nature of the rules of survival. It is a global optimization

algorithm. First of all, the path population is initialized. The path of individuals expressed as a series of dots in the path. Then the path of individual coding is equivalent to a chromosome in the biogenetic manipulation. Each point of the code referred to as gene. Then chromosome (several operators from string of individual coding) was selected, copied, crossover, mutation genetic operations. It immediately stops the evolution after the number of specific evolutionary. Finally it will output the current optimization of the individual. Therefore, we propose a new improved method, which enable the robot to adapt environmental capacity in a dynamic environment.

## 2.1. Strategic Choice

First, the environment can be modeled with the basic grid by using traditional methods. In order to facilitate the search to the optimal path, the grid granularity is small when using a basic grid in the environmental division of the optimization. And the obstacles will be accurate. But it will take a lot of storage space. Generated initial population of individual coding length is longer and the convergence speed is slow. Otherwise, population of individual coding length is shorter when the grid granularity is big. Accuracy of planning the path will decrease. Due to differences in the environment, the basic grid division is difficult to guarantee the grid granularity rationality. It results in the lack of adaptive capacity of genetic individual encoding. A blank area consist many genes with a basic grid to divide the entire region if the obstacle is small in an environment. There will be a lot of redundant and useless information in individual coding. This not only increases the storage space, but also the efficiency of genetic algorithm reduce in carrying out operations such as genetic operators. Therefore, zooming in the environment division can adaptively change according to the number of obstacles by the number of obstacles in the environment for inspiration. It can enhance the efficiency of the algorithm the grid granularity.

Based robot movement space is a two-dimensional plane. It denoted by *SE*. Starting point is *Start(S)* and goal point is *End (E)*. The optimization of the robot path planning guidelines can get the shortest path for the safe to avoid the collision in this paper. It means quickly to find a shortest path and avoid obstacles from *S* to *T*. Therefore, our definite location of the robot at some point in the *SE* plane is $P(x, y)$. Minimum step size of the robot is $\theta_{\min}$, Maximum step size is $\theta_{\max}$. It is shown in **Figure 2**.

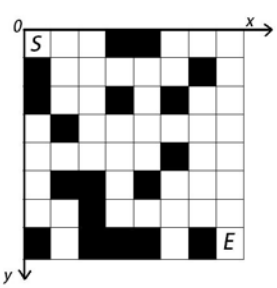


**Figure2.** Grid division

As shown, a flat area to the *SE*, the abscissa is x and the vertical axis is y. The minimum step size of the robot is divided into *M* rows and *N* columns by dividing the entire region. It divided as **Eq.1**,

$$\begin{aligned} N &= x / \theta_{\min} \\ M &= y / \theta_{\min} \end{aligned} \tag{1}$$

So the entire *SE* region initially divided into *M* rows and *N* columns. There are many genetic make up of the individual coding when the obstacle is less divided in a blank area of this environment. There will be a lot of redundant and useless information so that the storage space will increase. So the efficiency of the genetic algorithm will become bad during the genetic operator and other operations.

The proportion of obstacles share grid number and the total grid number is *L*. Then we defined the threshold *a* (it is obtained by the large number of environmental analysis experience in statistical value). It judge to zoom in or out of the grid by *a*.

*L* value obtained on the environment by the basic grid divided. Fewer obstructions in the environment should be enlarged the grid when *L* <*a*. We increased granularity to reduce the individual coding length and improve the convergence speed. The grid is enlarged by operating as follows.

At first, we use the minimum step $\theta_{\min}$ to divide *SE* into $M * N$ by basic grid. Then we calculate the *L*. Its grid is enlarged when *L*> *a* or using $2\theta_{\min}$ steps of long to divide plane. This cycle until it reaches the maximum step $\theta_{\max}$. There are 64 dominant genes in **Figure 3(a)**. At the same time, we enlarge the grid as **Figure 3(b)**. The dominant gene is 16.

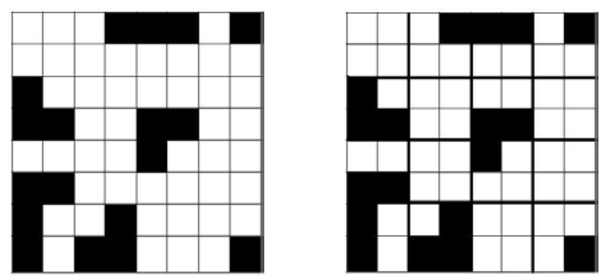

**Figure3. (a)** First division. **(b)** Second division.

Robot safety move forward in this case. Locking phenomenon appears when a position is not move forward. Grid granularity is relatively too much so that just narrow grid division no longer adapt to the current location of the current moment. Therefore, we need to timely adjust the grid. Let the grid enlarge and shrink throughout the whole path by the cycle repeated.

In this way, we can continue to zoom in and out of the grid when it is in the local area or obstacles in the move. And we can find a suitable particle size to make the algorithm efficiency.

## 2.2. Optimization of Obstacles

The only condition need to consider is an obstacle in robot path planning. Thus, the obstacle detection and analysis is extremely important. Therefore, we will have several common obstacle

distribution of optimization to improve the computational efficiency of the algorithm. The obstacles distribution shown in **Figure 4(a)** when the grid divided the whole region[5].

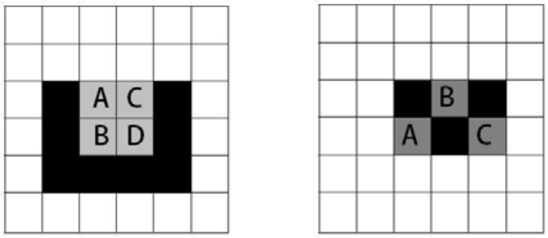


**Figure4. (a)** Situation A. **(b)** Situation B.

It detect point *B* and point *D* when the robot at point *A* or point *C*. *B* and *D* are treated as obstacles if it obtained position *B* and *D* are obstacles around. Further, the *A*, *B*, *C* and *D* are treated as obstacles. *A*, *B* and *C* are treated as obstacles when divided into grid after the obstacle distribution appears in **Figure 4(b)**. You can improve the computational efficiency of the algorithm for default processing of such obstacles. Finally, it avoids the locking phenomenon of the robot.

## 3. OPTIMIZATION OF FITNESS FUNCTION

Fitness function determines the pros and cons of the individuals in a population in genetic algorithm. The group may appear a small number of fitness excellent individual in the early stages of the algorithm runs. The final of these individuals may be filled with the whole group. Then the crossover operation of new individuals will have no effect. It will make the diversity of the population weakened. It also results in a genetic algorithm advance convergence to a local optimal solution. Therefore, avoiding premature convergence of the fitness function is necessary. It reduces the fitness of a higher individual and differences between other individuals. Finally it maintains the diversity of the population.

In order to accurate length measurement, we measure the distance of two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ in two-dimensional plane by using the Manhattan distance formulas such as **Eq.2**.

$$d_{12} = |x_1 - x_2| + |y_1 - y_2| \quad (2)$$

**Figure 5** shows the Manhattan distance between the location *P1* (0,0) and *P2* (2,1) is 4. And the Manhattan distance between *P1* (0,0) and *P3* (1,3) is 5.

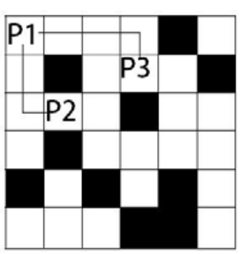


**Figure5.** The Manhattan distance

In this paper, we use **Eq.3** of the Manhattan distance to calculate the length of the robot path. $P_{t+1}$ is the point which is after the robot turned to a location. It is shown in **Figure 6(a)**. The Manhattan distance measurement can reduce the number of operations compared with the ordinary Euclidean distance. For example, the calculate times of the Euclidean distance from $P_1$ to $P_2$ is 5 in **Figure 6(b)**. And the Manhattan distance calculate times is only 2.

$$d(P) = \sum d|P_i P_{t+1}| \quad (3)$$

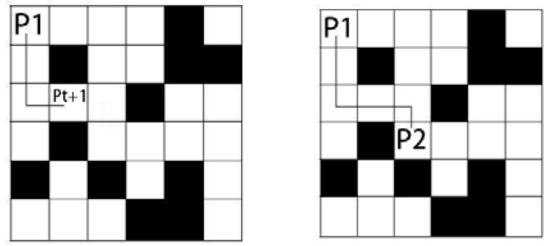


**Figure6. (a)** Location $P_{t+1}$ **(b)** Calculate times

The path length $L_{\mathrm{i}}$ is treated as the evaluation index. There is the final population to the index decreases in the direction of evolution. The fitness function of **Eq.4** is below,

$$L_{\mathrm{i}} = \lambda / L(P_i) \quad (4)$$

$d(P_i)$ is the individual's path length and $\lambda$ is the value of the lessons learned after the experiment.

The groups are increasingly concentrated in the later stages of the genetic algorithm. The difference between individuals is smaller and smaller. Competitiveness between individuals is also reduced. It can make the process of evolution lose their competitiveness. It becomes the random selection process. Therefore, we propose a nonlinear fitness to improve the genetic algorithm because of the lack of linear fitness. Then we construct the fitness function with the nonlinear evolution of the algebraic dynamic adjustment in minimization values for the shortest path problem as **Eq.5**,

$$L^*(X) = \frac{\left|\sqrt[\mathrm{m}]{\mathrm{n}}\right|}{H(X)} \quad (5)$$

$L^*(X)$ is non-linear fitness function. $H(X)$ is the objective function after the constraint handling. $m = \ln N + 1$, *N* is set in accordance with the complexity of the problem of evolution generation. *n* is the current evolution of generation.

Thus, according to the mathematical transformation of the fitness function as follows **Eq.6**,

$$L^*(X) = \frac{\left|\sqrt[\mathrm{m}]{\mathrm{n}}\right|}{\lambda/\mathrm{L}(P_i)} \quad (6)$$

The evolution generation is 100 - 500 by studying. It is shown that this evolution generation within the scope of the fitness function can be dynamically adjusted individual fitness by **Eq.5**. And it is also shown that the evolution generation and the length of the individual bit string are related. With taking into account the cost of the individual bit string length and algorithm run, the maximum evolution generation *N* is set to 200[6].

## 4. ALGORITHM PROCESSES AND SIMULATION ANALYSIS

Algorithm flow is shown in **Figure 7** by the design of the optimization.

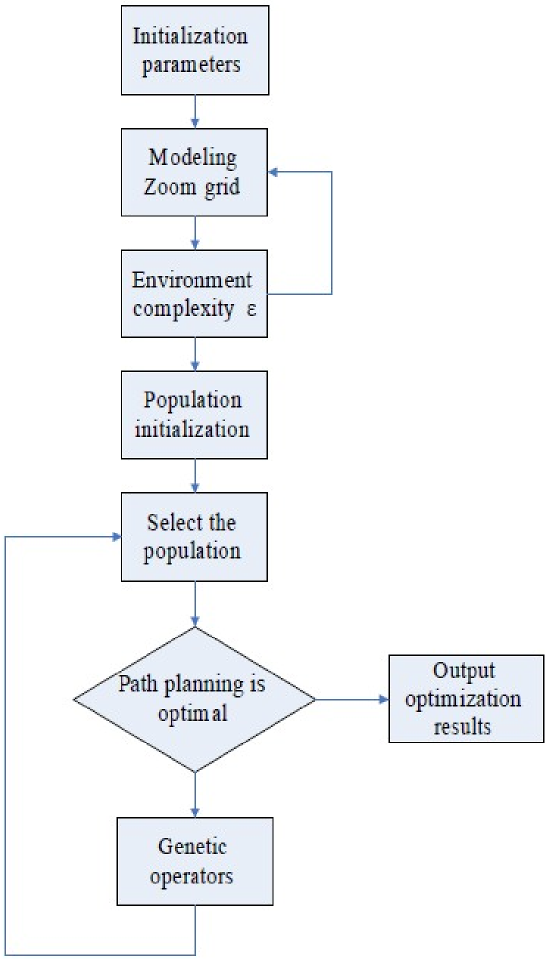


**Figure7.** Path planning flowchart

According to the conditions, parameters are initialized. And the number $N$ of evolution is selected to 200. There is the first division in environment with robot step. The obstacles of the proportion number $L$ determine whether amplification grid or not. And there is the division by $2\theta_{\min}$ until the arrival to $\theta_{\max}$. Grid is divided again with the appropriate environment complexity when there is no locking phenomenon. Grid of dynamic partition can improve the efficiency. And $\varepsilon$ is set to 0.01. Then the initial scale of populations is 40. Individual coding recorded as $P(0)=(P_1(0),P_2(0),P_3(0)\cdots,P_{40}(0))$ by the initial population.

Roulette method for operator selection is used in this case. The individual of populations is fitness function evaluation firstly. Then the highest individual directly copied to the next generation. Individual's choice probability and fitness value is proportional to. That means the greater the individual fitness value is, the higher the probability of the selected. Assuming that the number of population is $n$, the adaptability of a individual $i$ is $P_i$. So the selected probability is **Eq.7**,

$$P_i = \frac{f_i}{\sum_{i=1}^{x} f_i} \quad (7)$$

Individual coding is genetic manipulation if the path is not optimal. The operator between individuals in the same generation with a certain probability to exchange part of the gene to generate new populations. In addition, there is also a certain probability of mutation in the population. It can produce a new population through crossover and mutation of the gene. This provides the conditions for the next evolution. The good gene which is extracted in the original selection of the operator will copy to the next generation, with making the population more excellent.

In order to verify the effectiveness and feasibility of the proposed path planning method, the computer simulation is experimented by using Matlab7.0. The best fitness and average generation usually cannot get the minimum value. The initial population size is 40 through comparison and selection. Crossover rate is 0.5 and individual coding mutation rate is 0.01. According to experiences, $\lambda$ is 0.65. Environmental complexity $\varepsilon$ is 0.01. Maximum genetic evolution generation $N$ is 200. Each parameter value is selected for path planning simulation. Experiment randomly generated different obstacles. And it determines the starting point and goal point. The experimental results are satisfactory. As shown in **Figure 8(a)** for the genetic algorithm path and **Figure 8(b)** shows the improved genetic algorithm path.

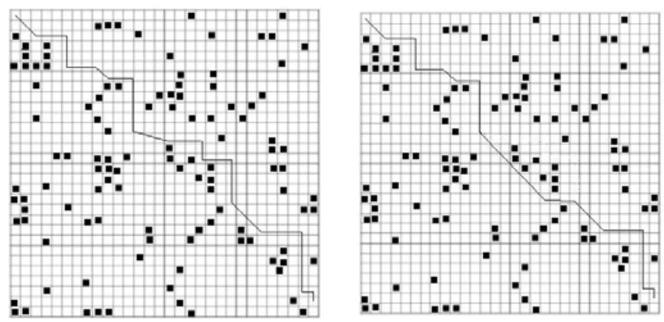

**Figure8. (a)** GA path planning **(b)** MGA path planning

In these circumstances, we respectively use the ant colony algorithm (AOC), the ordinary genetic algorithm (GA) and the improved genetic algorithm (MGA) path planning. Then the indicators were compared. As shown in **Table 1**, it can be seen that the improved genetic algorithm get good results.

**Table 1.** Comparison Chart

| Algorithm | Path Length | Time |
|---|---|---|
| AOC | 45.325 | 1.22 |
| GA | 46.255 | 2.13 |
| MGA | 41.523 | 0.95 |

As shown in **Figure 9** it is the two adaptation of the comparison of the results. We can see that the nonlinear fitness function genetic algorithm convergence and the operation efficiency is improved obviously.

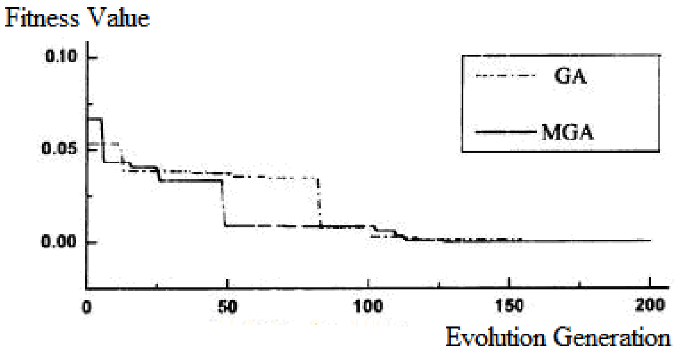


**Figure9.** The comparison of fitness function

## 5. CONCLUSION

This paper presents the dynamic scaling grid environment a novel modeling method in grid environment. Randomly generated by the genetic make up of the individual chromosomes(individual coding) may have many invalid path after basic grid division plane when there are many obstacles in the grid environment, with making the algorithm's efficiency. This method can improve the effectiveness of the initial population. And it also increases the effectiveness of every gene in individual coding. According to the environment immediately enlarged or reduced grid, it can achieve modeling optimal with grid size adjustment, which makes the algorithm efficiency. Aiming at simple genetic algorithm with linear fitness shortcomings, this paper also proposes the nonlinear fitness function genetic algorithm. The simulation results show that the improved genetic algorithm is feasible and effective. It also shows that the nonlinear fitness function of genetic operation is more adapt to the process of evolution.

## REFERENCES

[1] Derek J Bennet, Colin R McInnes. Distributed control of multi-robot systems using bifurcating potential fields. Robotics and Autonomous Systems,(2010),pp.256-264

[2] DENG Wang-yu, ZHENG Qing-hua, CHEN Lin, et al. Research on extreme learning of neural networks. Chinese Journal of Computers (2010), pp. 279-289.

[3] Garcia M A Porta, Montiel Oscar, Castillo Oscar, et al. Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation. Applied Soft Computing(2009)

[4] Wang Qiang, Yao Jin, Wang Jinge. Based on genetic algorithm of mobile robot a path planning method. Journal of Harbin Institute of Technology, (2004)

[5] LIU Chang-an, CHANG Jin-gang, LIU Chun-yang. Path planning for mobile robot based on an improved probabilistic roadmap method. Chinese Journal of Electronics, (2009), pp.395-399.

[6] Cai Zixing, Peng Zhihong. A new path encoding mechanism in path planning of mobile robot. Robot. (2001), pp.230 – 233